



# Unraveling robustness of deep face anti-spoofing models against pixel attacks

Naima Bousnina<sup>1</sup> · Lilei Zheng<sup>2</sup> · Mounia Mikram<sup>3</sup> · Sanaa Ghouzali<sup>4</sup> · Khalid Minaoui<sup>1</sup>

Received: 17 February 2020 / Revised: 20 August 2020 / Accepted: 7 October 2020 /

Published online: 27 October 2020

© Springer Science+Business Media, LLC, part of Springer Nature 2020

## Abstract

In the last few decades, deep-learning-based face verification and recognition systems have had enormous success in solving complex security problems. However, it has been recently shown that such efficient frameworks are vulnerable to face-spoofing attacks, which has led researchers to build proficient anti-facial-spoofing (or liveness detection) models as an additional security layer. In response, increasingly challenging and tricky attacks have been launched to fool these anti-spoofing mechanisms. In this context, this paper presents the results of an analytical study on transfer-learning-based convolutional neural networks (CNNs) for face liveness detection and differential evolution-based adversarial attacks to evaluate the efficiency of face anti-spoofing classifiers against adversarial attacks. Specifically, experiments were conducted under different use-case scenarios on four face anti-spoofing databases to highlight practical criteria that can be used in the development of countermeasures to address face-spoofing issues.

**Keywords** Face liveness detection · Spoofing attacks · Convolutional neural networks · Differential evolution · Deep learning

## 1 Introduction

Facial biometrics consistently outperform other biometric modalities in a wide range of daily applications in terms of their reasonable recognition cost, convenience, and high levels of performance. As examples of the applicability of the approach, Lenovo, Asus, and Toshiba laptops now come with built-in face authentication webcams [38] and the Unique Identification Authority of India (UIDAI) facial recognition system is used to identify Indian

---

✉ Naima Bousnina  
naimabousnina2@gmail.com

residents [56]. As the general public becomes increasingly acquainted with facial authentication systems, their loopholes are being explored. The human face can be easily acquired and duplicated by attackers who can obtain facial images or videos from social networks and use them to generate artificial models, which can then be used to deceive face authentication systems in an attack mode referred to as face spoofing. This presents a challenge to authentication mechanisms, which, in addition to delivering high recognition performance, must be able to differentiate between live and fake users.

Broadly speaking, spoofing attacks involve a series of manipulative actions with the goal of gaining illegitimate access to biometric authentication systems by presenting an artificial, rigged version of original biometric data to a system sensor. Spoofing attacks are also known as presentation attacks and defined in the first part of the ISO/IEC 30107 standard as “*Presentation of an artefact or human characteristic to the biometric capture subsystem in a fashion that could interfere with the intended policy of the biometric system*” [19]. Face-spoofing attacks can be classified into three categories: (1) *print attack-2D*, involving the use of printed photos or screen-displayed images; (2) *replay attack-2D*, involving the playing of video in front of a system sensor; and (3) *mask attack-3D*, involving the use of 3D masks, facial cosmetic makeup, or even plastic surgery. To guarantee a strong degree of security against face-spoofing attacks, it is necessary to equip authentication frameworks with facial anti-spoofing models, an approach also referred to as presentation attack detection or liveness detection modeling. A number of approaches to solving the spoofing attack research problem have recently been developed, as witnessed by the rising number of studies in which biometric authentication system sensitivity has been evaluated and new defense techniques have been explored [12, 16, 22, 28, 42, 49]. Several workshops and competitions have been organized [8, 18], dedicated datasets have been developed [4, 7, 30, 40, 59, 61], and new projects have been launched to address this issue [53].

Inspired by the significant advances in the application of deep learning to a wide range of fields, including object detection [43] and recognition [24] and speech recognition [44], convolutional neural network (CNN)-based face anti-spoofing algorithms have had phenomenal success in achieving state-of-the-art results against diverse types of spoofing [13, 27, 60]. As an illustration, the 13 teams that made it to the final round of the competition hosted by ChaLearn at Computer Vision and Pattern Recognition (CVPR) all adopted CNN-based solutions [33]. The underlying concept behind CNN-based face anti-spoofing is to directly extract the most pertinent end-to-end biometric features from an original dataset to catch variants of both well-known and unknown adversarial attacks.

Despite its considerable success, many CNN-based face liveness detection mechanisms are still vulnerable to adversarial attacks. Recent studies have demonstrated the ease with which such mechanisms can be bypassed by applying slight, imperceptible perturbations to input samples [9, 14, 35, 52]. Based on the access level to the attacked model, deep-learning-based adversarial attacks can be classified into two categories: (1) *white-* and (2) *black-box attacks*. In the former, full access to the attacked model’s parameters and architecture is granted while, in the latter, access is given only to the attacked model’s inputs and outputs [39]. Since Szegedy et al. [52] demonstrated that even well-performing deep neural networks (DNNs) are vulnerable to adversarial attacks, an extensive body of research has continued to find new adversarial attack methods. For instance, a number of first-order algorithms for producing adversarial samples have been suggested, namely, fast gradient sign method (FGSM) [14], iterative gradient sign method (IGSM) [25], projected gradient descent (PGD) [26], and Carlini & Wagner attacks [5]. At the same time, related defense mechanisms have been proposed to bypass such attacks.

In the context of this ongoing struggle between anti-facial-spoofing systems and spoofing attacks, the current paper describes an analytical study on adversarial attacks to deep-learning-based face liveness detection models. The primary goal of this study was to assess how deep face anti-spoofing mechanisms behave in the presence of spoofing attacks. The main contributions of this work are as follows:

- We present an experimental setup for emulating differential evolution-based adversarial attacks on transfer-learning-based CNNs for face liveness detection systems based on increasing the confidence with which fake faces can be classified as live faces.
- We produce an experimental assessment of more realistic circumstances in which real-world applications are emulated based on the results of tests on four different face anti-spoofing databases under different threat models and use-case scenarios.
- We assess the impact of various criteria related to both spoofing and anti-spoofing algorithms to improve the overall robustness of deep face anti-spoofing mechanisms. Based on this assessment, we attempt to outline how adversarial attacks reduce the contrast between live and fake faces, highlight common misconceptions, and derive complementary countermeasures that aid in constructing more flexible anti-spoofing frameworks and enhance their resistance to a wide variety of adversarial attacks.

The remainder of this paper is organized as follows. Section 2 summarizes the existing literature relating to deep-learning-based face spoofing and anti-spoofing approaches. Section 3 describes in detail the methodology we followed in our study and, finally, our experimental setup and results are presented and discussed in Section 4. Conclusions are drawn in Section 5.

## 2 Related works

In this section, we review the recent research literature on adversarial attacks and face liveness detection technologies. The primary focus is on CNN-based approaches.

### 2.1 CNN-based face anti-spoofing approaches

Recently, research on deep-learning-based face anti-spoofing begun to shift toward the use of CNNs. For instance, [37] suggested a performance evaluation of CNN-based face anti-spoofing using ResNet and Inception architectures in terms of architecture depth, learning rate, fine-tuning as opposed to training from scratch, and weight transfer as opposed to weight initialization. In [3], it was postulated that approaches involving treatment of the overall face either globally or in small batches diminishes algorithmic performance; based on this, a new CNN architecture to learn diverse local spoofing cues was proposed. This architecture is trained in two phases: first, each fraction of the network is trained on a specific facial area, which helps the model to catch varied spoofing cues from all parts of the face; then, the model generalization is enhanced by fine-tuning the overall model using fake and real images with the weights learned in the first step applied. In [54], the focus was on improving the generalization capacity across various databases; more specifically, a CNN-based framework was used to effectively adjust new domains with sparsely labelled target domain data for facial anti-spoofing. G. Goswami et al. [15] developed a deep network method to carry out adversarial attack detection and mitigation and used it to develop an image processing-based adversarial attack scheme and an automatic detection procedure to fend off such attacks. Finally, a scheme for adversarial attack mitigation was suggested to

improve the overall robustness of DNN-based face recognition. The suggested approaches were evaluated using quasi-imperceptible disfigurement approaches, including elastic-net attacks to DNNs (EAD), universal adversarial perturbation, DeepFool, and l2 distance, and built using cross-database and cross-attack scenarios. In [17], a 3-D face anti-spoofing algorithm was developed using a hypergraph CNN (HGCNN). In [1], a novel face liveness detection technique based on the fusion of two CNN architectures was introduced; in this approach, the first CNN structure was used to extract local features, while the latter was used to extract holistic features. Another two-stream CNN architecture was developed in [21] through the combination of a CNN with auxiliary supervision with a generative adversarial network (GAN)-like discriminator to achieve facial de-spoofing. The primary concept behind this approach was to reciprocally decompose spoofed faces into spoof noise and live faces and then apply classification based on the spoof noise.

A number of approaches have combined CNNs with other architectures and methods [30, 32, 55, 57]. For example, in [55] a common CNN-LSTM network to detect face attacks across video frames was suggested. To achieve this goal, highly discriminative features of video frames were extracted via CNNs and then a long short-term memory (LSTM) architecture was used to capture the temporal dynamics of the videos. In [41], a robust feature representation scheme that combined deep texture features and eye-blink cues for facial anti-spoofing was suggested. The proposed approach involved learning deep texture features from aligned face images and unaligned video frames via CaffeNet and GoogLeNet and then employing the frame differences to detect eye-blinking.

Other studies have focused on detecting adversarial attacks using new architectures. For instance, [11] introduced a multi-cue integration framework for face anti-spoofing under which shearlet-based image quality features and dense optical flow are used to, respectively, develop an image quality-based liveness feature and extract motion-based liveness features. A bottleneck-based Autoencoder feature fusion methodology is then used to merge the different liveness detection characteristics to arrive at efficient decisions. In [31], a deep tree network for acquiring features and revealing unknown spoof attacks hierarchically was adopted. In [58], facial depth and temporal motion approaches were fused to distinguish between fake and live faces using (i) *single-* and (ii) *multi-frame* modules comprising depth-supervised neural network architectures with optical flow guided feature block and convolution gated recurrent units. The single-frame module estimates depth maps from individual frames, while the multi-frame module is used to merge short- and long-term motion extractors.

## 2.2 Adversarial attacks to face anti-spoofing algorithms

To bypass face liveness detection systems, a number of tricky adversarial attacks have been introduced. An example of this is the targeted white-box adversarial attack presented in [48] as a tool for fooling face recognition systems. The proposed attack uses the attentional adversarial attack generative network ( $A^3GN$ ) approach to produce adversarial samples identical to original face images. Unlike conventional GANs, this architecture applies facial recognition as a third component of the conflict between generator and discriminator to enable the efficient imitation of a target individual. In [2], a new approach to simulating adversarial samples by solving a constrained optimization problem using an adversarial generator network was proposed. The principle underlying the proposed strategy of generating small distortions to an input image to deceive face spoofing was tested on a faster R-CNN-based face detector. Another intriguing study proved that facial recognition systems

could be fooled by building adversarial glasses [45]. M. Sharif et al. proposed an adversarial generative net architecture to misclassify DNN-based face classifiers by adding physical realizations of eyeglass frames [46]. Komulainen [6] proposed the concept of backdoor attack, in which poisoning samples are injected into a deep-learning-based authentication system with the goal of misleading it. In [23], 2- and 3-D face mask attack datasets constructed for the TABULA RASA project were used to prove the impact of mask attacks on 2-, 2.5-, and 3-D face recognition models. In [63], a novel adversarial attack scheme involving illumination of the attacker's face was discovered. Under this attack mode, infrared dots are projected onto specific positions on the face using minuscule infrared (IR) LEDs integrated into a cap, umbrella, or wig to mislead machine learning-based face recognition systems. Krizhevsky et al. [20] proposed a black-box adversarial attack to produce adversarial distortions based on the output of a differential evolution algorithm. The approach is designed to manipulate only a few pixels of the input face image to misclassify CNN-based classifiers. S. Moosavi-Dezfooli et al. [36] suggested a DeepFool approach to efficiently calculate distributions that deceive deep networks and compared the robustness of various deep network classifiers against adversarial perturbations.

### 3 Methodology

Adversarial attacks are used to minimize the resistance of liveness detection systems. In a successful case, if an adversarial sample is presented to a system sensor, the authentication attempt will be viewed as a genuine identity and will be accepted. Playing the role of a neutral referee to assess the vulnerability of face anti-spoofing measures against presentation attacks, we carried out an analytical study on adversarial attacks to deep learning-based face liveness detection systems. Our goal was to evaluate and discuss the effectiveness of face anti-spoofing models when confronted with spoofing attacks. Based on this evaluation, we attempted to highlight the various criteria for constructing more flexible anti-spoofing frameworks and improve their resistance to a wide scope of adversarial attacks. To accomplish this, we investigated the impact of differential evolution-based adversarial perturbation on transfer-learning-based convolutional neural networks for face anti-spoofing. Specifically, we experimentally evaluated how well such approaches reject adversarial samples under different use-case scenarios using four different anti-face-spoofing databases. In the following subsections, we describe the face spoofing and anti-spoofing approaches assessed in our study.

#### 3.1 Transfer-learning-based CNNs for face anti-spoofing

Following the approach used in [34], the proposed face anti-spoofing classifier was built using transfer-learning based on a pre-trained VGG16 CNN structure [47]. Within the VGG16 network, there are up to 13 convolutional (Conv) layers of  $3 \times 3$  kernels and three fully connected layers of size 4,096, 1,000, and 1,000, respectively. The VGG16 architecture is built using identical padding (Zero Pad) and max-pooling (Max Pool)  $2 \times 2$  window and stride 2 layers. The rectifier linear unit activation function is applied to the output of each convolutional and fully connected layer.

Figure 1 shows the proposed face anti-spoofing network (FASNet). With the exception of the top layers, the architecture replicates the VGG16 structure; specifically, the 13 convolutional layers (highlighted in grey) are retained, while the three fully connected layers are

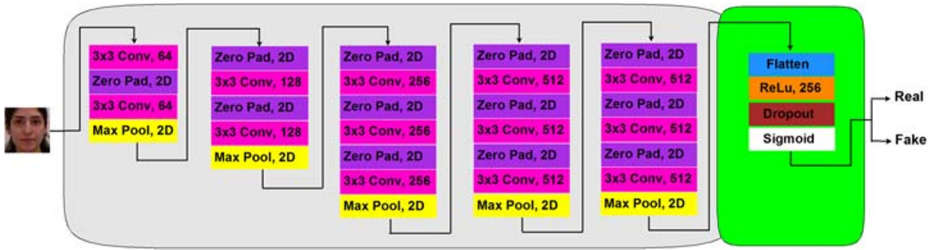


Fig. 1 Face Anti-Spoofing Network (FASNet) architecture

omitted and replaced by two fully connected layers of sizes 256 and 1, respectively (highlighted in green), to enable the binary classification required to carry out the face liveness detection task. The first fully connected layer is followed by one dropout layer to avoid overfitting. To enable finer classification, the softmax decision function is replaced by a sigmoid function. Transfer learning is carried out by freezing the weights of the first seven layers and fine-tuning the weights from the eighth up to the top layers through backpropagation.

### 3.2 Differential evolution algorithm

Differential evolution is a global optimization and stochastic direct search algorithmic approach [50] in which linear combinations are applied instead of conventional crossover and mutation operators to generate new solutions.

For a candidate population  $p_{n,i}^g = [p_{n,1}^g, p_{n,2}^g, p_{n,3}^g, \dots, p_{n,D}^g]$  represented by an  $N$ -dimensional vector, where  $g$  is the generation and  $D$  is the number of variables, we can define  $f$  as the fitness function of a candidate solution. The differential evolution algorithm comprises four steps: initial population, mutation, recombination, and selection.

#### – Step 1: Initial Population

To start with, the initial population is randomly generated between the upper and lower bounds as follows:

$$p_{n,i} = p_{n,i}^L + rand() * (p_{n,i}^U - p_{n,i}^L) \quad (1)$$

where  $p_{n,i}^L$  and  $p_{n,i}^U$  are the lower and upper bounds of the variable  $p_{n,i}$ , respectively,  $i = 1, 2, 3, \dots, D$  and  $n = 1, 2, 3, \dots, N$ .

#### – Step 2: Mutation

In the mutation stage, the current generation is perturbed by scaling the difference between randomly selected population candidates,  $p_{r2n}^g$  and  $p_{r3n}^g$ ; the scaled difference is then added to a third randomly selected population candidate to generate a new donor candidate  $v_n^{g+1}$ :

$$v_n^{g+1} = p_{r1n}^g + F * (p_{r2n}^g - p_{r3n}^g) \quad (2)$$

where  $F \in [0, 1]$  is the scale parameter and  $r1n$ ,  $r2n$ , and  $r3n$  ( $r1n \neq r2n \neq r3n$ ) are random indices of the parent population.

#### – Step 3: Recombination

In the recombination phase, a trial candidate  $u_{n,i}^{g+1}$  is evolved using both  $p_{n,i}^g$  and  $v_n^{g+1}$  as follows:

$$u_{n,i}^{g+1} = \begin{cases} v_{n,i}^{g+1} & \text{if } rand() \leq C_p \text{ or } i = I_{rand} \quad i = 1, 2, 3, \dots, D \text{ and} \\ p_{n,i}^g & \text{if } rand() > C_p \text{ and } i \neq I_{rand} \quad n = 1, 2, 3, \dots, N \end{cases} \quad (3)$$

where  $I_{rand}$  is a random integer in  $[1, D]$  and  $C_p$  is the recombination probability.

– **Step 4: Selection**

Finally, in the selection stage each generated child  $u_{n,i}^{g+1}$  is compared with its corresponding parents  $p_{n,i}^g$ :

$$p_n^{g+1} = \begin{cases} u_{n,i}^{g+1} & \text{if } f(u_{n,i}^{g+1}) < f(p_n^g) \\ p_n^g & \text{Otherwise } n = 1, 2, 3, \dots, N \end{cases} \quad (4)$$

and the worse-performing parents are replaced with better children.

### 3.3 Differential evolution-based adversarial attack

To deceive the FASNet system, the one-pixel adversarial attack introduced in [51] was utilized. The concept underlying one-pixel attacks is to generate adversarial samples by locating the pertinent pixels in the input image and their corresponding strengths of perturbation. Unlike most conventional adversarial attacks, the one-pixel attack focuses on distorting only a few pixels with varying manipulation strength, rather than distorting all pixels with an overall constraint on the strength of distortion.

Generating adversarial samples can be considered to represent an optimization problem with constraints. For an output of the image classifier,  $f_t(\mathbf{z})$ , where  $\mathbf{z} = (z_1, z_2, \dots, z_N)$  is the N-dimensional vector representation of an input image correctly classified as class  $t$ , the adversarial samples can be defined using the following equation:

$$\begin{aligned} \mathbf{z}' &= \mathbf{z} + e(\mathbf{z}) \\ \{\mathbf{z}' \in R^N \mid \arg \max_i (f_{adv}(\mathbf{z}'))_i &\neq \arg \max_j (f_t(\mathbf{z}))_j\} \end{aligned} \quad (5)$$

where  $e(\mathbf{z}) = (e_1, e_1, \dots, e_n)$  is a very small additive adversarial perturbation related to  $\mathbf{z}$  and the target class  $adv$ . The goal of using adversary samples is to reach an optimized perturbation  $e(\mathbf{z})^*$  that raises the confidence of the target class and decreases the confidence of the original class:

$$\begin{aligned} &\text{maximize}_{e(\mathbf{z})^*} f_{adv}(\mathbf{z}') \\ &\text{Subject to } \|e(\mathbf{z})\|_0 \leq d \end{aligned} \quad (6)$$

where  $d$  is a small number that is  $d = 1$  for a one-pixel attack.

To accomplish this, differential evolution as a population-based optimization algorithm is based. This algorithm follows a very limited scenario in which only the probability labels, with no inner information regarding the attacked model, are required; this categorizes the attack as a black-box adversarial attack.

In the context of the one-pixel adversarial attack, we define  $\mathbf{x}=(x,y,r,g,b)$  as a one-pixel perturbation in which  $x$ ,  $y$  and  $r$ ,  $g$ ,  $b$  are the pixel coordinates and RGB channel, respectively. In a similar manner, multiple perturbations can be presented as a concatenation of several tuples as follows:

$$\begin{aligned} X &= (\mathbf{x}_1, \mathbf{x}_2, \dots) \\ &= (x_1, y_1, r_1, g_1, b_1, x_2, y_2, r_2, g_2, b_2, \dots) \end{aligned} \quad (7)$$

As illustrated in Fig. 2, the attack is generated by randomly initializing an N-dimensional population of perturbations  $P = (X_1, X_2, X_3, \dots, X_N)$ . Following,  $N$  new mutant children are created using:

$$X_i = X_{r1} + F \times (X_{r2} + X_{r3}) \quad (8)$$



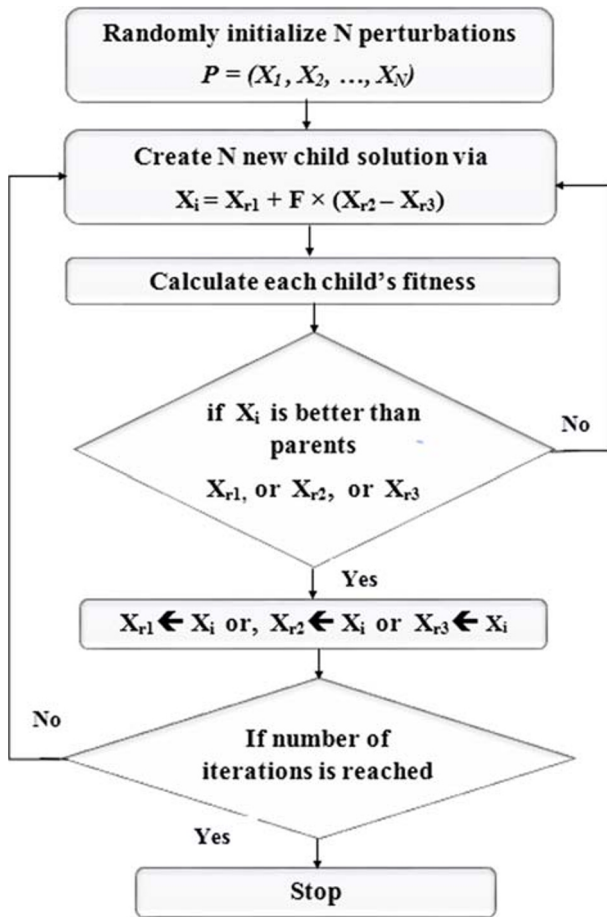


Fig. 2 Flowchart of one-pixel adversarial attack algorithm

where  $r1, r2, r3 (r1 \neq r2 \neq r3)$  are random indices of the parent population  $P$  and  $F$  is the scale parameter, which is set to 0.5. Once the new generation has been created, a fitness comparison is carried out between each child and its corresponding parents according to the population index, and the best-performing individuals survive to the next iteration. This procedure is repeated for several iterations until the stopping criterion is satisfied. The perturbation corresponding to the best fitness is then taken as the best pixel for manipulation in the input image.

### 3.3.1 Face anti-spoofing datasets

Experiments were conducted using four publicly available face anti-spoofing datasets, namely 3D Mask Attack Database (3DMAD) [10], Replay-Attack [7], CASIA [61], and ROSE-Youtu [29]. As the datasets comprise video clips while FASNet is a 2-D structure, a two-step pre-processing procedure, in which several frames were uniformly extracted from each video and then the Multi-Task CNN (MTCNN) was employed for face location [62], was carried out.



- **3DMAD:** This dataset comprises 255 videos of 17 individuals. All frames are registered via Kinect and characterized by a depth image, corresponding RGB image, and manually annotated eye position. The registrations are carried out using frontal views with neutral expressions. Each record is gathered in three different sittings: in the first two, real-access samples are registered with a time delay of two weeks between acquisitions; in the third, a 3-D mask attack is obtained.
- **CASIA:** This dataset comprises 50 genuine subjects for whom there are three genuine and nine fake videos apiece. Thus, the global dataset contains 600 video clips. Three types of attacks printed photographs with the eyes cut out, video attacks, and warped printed photographs are considered. Each subjects is recorded using three types of cameras: low, average, and high quality.
- **Replay-attack:** This dataset is produced at the Idiap research institute with the specific goal of developing anti-spoofing algorithms. Each video in this dataset is generated by placing a real individual in front of a built-in webcam or by re-recording a video or photo of the individual for at least 9-s under different lighting conditions. Depending on the type of device used, six protocols can be applied to produce a video attack: print, mobile (phone), high-definition (tablet), photo, video, or grand testing (a combination of all of the preceding).
- **ROSE-youtu:** This dataset comprises 4,225 videos of 25 subjects, with 150-200 ~10-s video clips per subject. The front-facing cameras of five cell phones—a Hasee smart-phone, Huawei smart-phone, iPad 4, iPhone 5s, and ZTE smart-phone are used to collect the data. Genuine-face videos are recorded under five different illumination conditions, and three adversarial attacks printed paper, video replay, and masking attacks are taken into consideration.

4 Experimental analysis

In this section, we describe the experimental setup employed in the study and then discuss and evaluate the results of the experiments.

4.1 Experimental setup

To perform the experimental evaluation, training and test datasets comprising  $32 \times 32$  pixel images containing equal numbers of live and fake images were used. Table 1 lists the number of images used for each dataset. The approach used in the study was to perturb the fake images so that they were identified by the face liveness detection classifier as live images and then analyze the trade-off between the spoofing and anti-spoofing algorithms.

Table 1 Number of images used per dataset

Datasets	Training images		Test images	
	Live	Fake	Live	Fake
3DMAD	240	240	100	100
CASIA	360	360	540	540
Replay-Attack	1200	1200	800	800
ROSE-Youtu	2240	2240	2245	2245

To study the effects of attack on the face anti-spoofing systems, different criteria were assessed. In particular, different training step values were used to generate the respective face liveness detection models. Eight data augmentation (DA) techniques were applied, namely (1) *None (no DA)*, (2) *Width and Height Shift*, (3) *Shear Map*, (4) *Zooming*, (5) *Rotation*, (6) *Channel Shift*, (7) *Vertical and Horizontal Flipping*, and (8) *All DA*. In implementing differential evolution-based attacks, two additional criteria were considered the number of manipulated pixels per image and the number of attack iterations. The first parameter relates to the likelihood of a successful attack, as increasing the number of manipulated pixels will result in changes to more pixels in the image within a given time, while the second specifies the number of generations the differential evolution-based attack algorithm should run before giving up.

The initial population of the differential evolution algorithm was initialized using the Latin hypercube sampling method to minimize the number of iterations needed to achieve a reasonably accurate result. The fitness value was set as the probability (confidence) value of the original class. Once the confidence value for a fake face reached 0.5 or lower, it was classified by the anti-spoofing model as a live face. Unlike [34], in which an Adam optimizer was used, the default stochastic gradient descent (SGD)-based VGG16 optimizer was used. The FASNet model hyperparameters are listed in Table 2. The number of epochs used was varied experimentally based on the training steps.

The spoofing and anti-spoofing algorithms were both constructed using Keras with Tensorflow as a backend. All tests were carried out using a computer equipped with Ubuntu 18.04.2 LTS with a Tesla K80 GPU (12 GB).

## 4.2 Results and discussion

In this section, we evaluate the results of the experiments against five standard metrics: accuracy (ACC), half total error rate (HTER), precision, recall, and recall drop.

### 4.2.1 Overall results

To assess the anti-spoofing performance of the FASNet algorithm against those of other DA techniques, models for the 3DMAD, CASIA, and Replay-Attack datasets were generated using 50,000 training steps and for the ROSE-Youtu dataset using 100,000 steps. The number of steps used for the ROSE-Youtu dataset were doubled to account for the number of images used in both the training and test phases. Tables 3, 4, 5 and 6 summarize the performance of each model before and after an adversarial attack applying nine manipulated pixels over 200 attack iterations.

Most importantly, the results indicate that DA significantly improved the FASNet model's anti-spoofing performance on the more complex datasets. For example, on the simple 3DMAD dataset, the model without DA reached an accuracy of 100%, whereas, on the

**Table 2** FASNet model parameters

Parameter	FASNet Model
Learning Rate	$10^{-3}$
Momentum	0.9
Dropout	0.5
Batch Size	64

**Table 3** Anti-spoofing performance of FASNet model on 3DMAD dataset and recall drops caused by adversarial attack using nine manipulated pixels

3DMAD	Before attack				After attack	Recall drop(%)
DA used	ACC (%)	HTER (%)	Precision (%)	Recall(%)	Recall (%)	
None DA	100	0	100	100	81	19.00
Shift	100	0	100	100	98	2.00
Shear	100	0	100	100	77	23.00
Zooming	100	0	100	100	41	59.00
Rotation	100	0	100	100	0	100.00
Channel shift	100	0	100	100	75	25.00
Flipping	100	0	100	100	87	13.00
All	100	0	100	100	48	52.00

**Table 4** Anti-spoofing performance of FASNet model on CASIA dataset and recall drops caused by adversarial attack using nine manipulated pixels

CASIA	Before attack				After attack	Recall drop(%)
DA used	ACC (%)	HTER (%)	Precision (%)	Recall(%)	Recall (%)	
None DA	83.52	15.65	85.08	81.30	29.63	51.67
Shift	92.31	6.67	90.02	95.19	45.37	49.82
Shear	88.80	11.67	88.40	89.26	34.63	54.63
Zooming	92.78	7.41	92.56	93.15	29.07	64.08
Rotation	75.83	18.33	68.93	94.07	35.00	59.07
Channel shift	86.48	14.63	84.68	89.07	30.74	58.33
Flipping	91.67	8.89	89.89	93.89	37.04	56.85
All	96.02	3.80	97.15	94.81	27.41	67.40

**Table 5** Anti-spoofing performance of FASNet model on Replay-Attack dataset and recall drops caused by adversarial attack using nine manipulated pixels

Replay-Attack	Before attack				After attack	Recall drop(%)
DA used	ACC (%)	HTER (%)	Precision (%)	Recall(%)	Recall (%)	
None DA	98.81	1.38	99.37	98.25	73.50	24.75
Shift	99.81	0.12	99.75	99.88	84.44	15.44
Shear	98.44	1.25	98.99	97.88	71.88	26.00
Zooming	99.94	0.12	99.88	100.00	63.12	36.88
Rotation	98.88	1.19	98.63	99.12	77.12	22.00
Channel shift	96.81	3.12	96.99	96.62	73.25	23.37
Flipping	98.94	1.12	98.00	99.00	82.62	16.38
All	100	0	100	100	66.38	33.62

**Table 6** Anti-spoofing performance of FASNet model on ROSE-Youtu dataset and recall drops caused by adversarial attack using nine manipulated pixels

ROSE-Youtu	Before attack				After attack	Recall drop(%)
DA used	ACC (%)	HTER (%)	Precision (%)	Recall(%)	Recall (%)	
None DA	87.93	11.76	85.79	90.91	39.56	51.35
Shift	90.76	9.35	93.32	87.80	47.43	40.37
Shear	86.68	11.40	81.69	94.57	58.26	36.31
Zooming	91.98	8.02	92.74	91.09	39.27	51.82
Rotation	88.51	10.47	85.16	93.27	49.78	43.49
Channel shift	88.35	10.69	84.86	93.36	57.17	36.19
Flipping	89.78	9.40	78.30	93.10	54.93	38.17
All	91.92	8.57	93.52	90.07	36.35	53.72

CASIA dataset, the model without DA achieved an accuracy of 83.52%, which is much worse than the 96.02% accuracy achieved by the model applying all DA techniques.

The second important finding is that using an adversarial attack with nine manipulated pixels can significantly reduce the FASNet model's ability to recall all spoofing faces. For example, on the Replay-Attack dataset, the FASNet model with DA was able to correctly prevent spoofing with 100% accuracy; when the attack manipulated nine pixels of a fake facial image, however, 33.62% of the fake faces were wrongly identified as actual faces. The same results were obtained on the ROSE-Youtu dataset, in which 53.72% of the fake faces were wrongly classified as genuine faces.

#### 4.2.2 Effect of the number of manipulated pixels

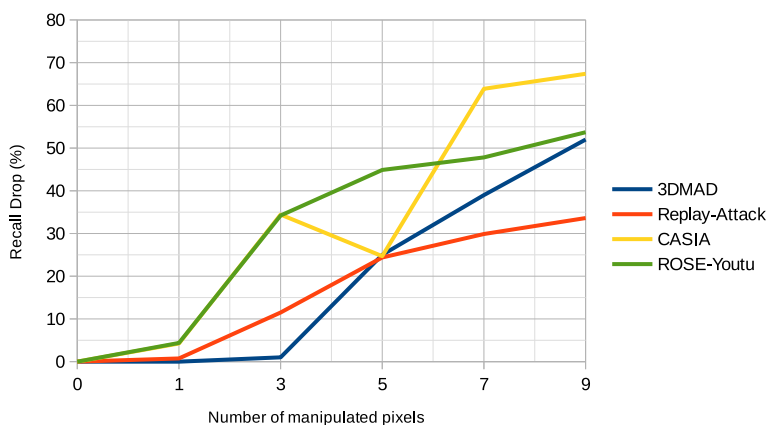
Next, the effect of the number of pixels manipulated by the adversarial attack on perturbation effectiveness was assessed. Specifically, different adversarial samples were generated by changing the number of manipulated pixels per image within a range from one to nine while applying a constant 200 attack iterations. Figure 3 shows the variations in recall drop with the number of number of changed pixels per image over the four datasets.

It is seen that increasing the number of pixels manipulated leads to a more significant recall drop, which demonstrates the impact of the number of changed pixels in increasing the likelihood of a successful attack. Given these results, we believe that even well-trained models can be relatively vulnerable to such perturbations under strict constraints.

#### 4.2.3 Effect of the number of attack iterations

We then measured the adversarial attack performance as a function of the number of attack iterations. To do so, the pixel perturbation process was applied under a varying number of iterations to specify the number of generations that the attack algorithm should run before giving up. Figure 4 shows the recall drop as a function of number of attack iterations with the number of manipulated pixels held constant at nine.

It is seen that, on all of the datasets, increasing the number of attack iterations increases the chance that the attack will deceive the anti-spoofing model, as the recall drop grows with the number of iterations. When the number of attack iterations is less than 300, increasing



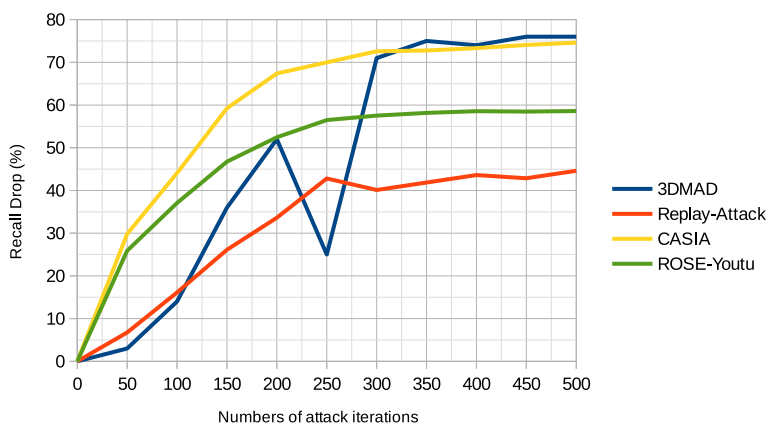
**Fig. 3** Number of attacked images as a function of number of manipulated pixels per image for the 3DMAD, CASIA, Replay-attack, and ROSE-Youtu datasets

the number of iterations has a more significant effect on the recall drop; in general, the most significant recall drop is achieved by the time the number of attack iterations reaches 300.

#### 4.2.4 Effect of the number of training steps used by the anti-spoofing model

Finally, we investigated the effect of the number of training steps used by the anti-spoofing model. The results listed in Tables 7 and 8 show the anti-spoofing performance obtained at various numbers of training steps on the CASIA and Replay-Attack datasets, respectively.

It is evident from both tables that increasing the number of training steps does not make a model more resistant to adversarial attack, as the post-attack recall variations do not decrease as the number of steps increases. For instance, the post-attack recall on the Replay-Attack dataset is 50.75% when the model is trained using 2,500 steps; this is increased to 66.38% when the number of steps is increased to 50,000. Similar results are obtained on the



**Fig. 4** Recall drop as a function of number of attack iterations with nine manipulated pixels per attack

**Table 7** Recall variation at different numbers of training steps for the anti-spoofing model on the CASIA dataset. The numbers of manipulated pixels and attack iterations are held constant at nine and 200, respectively

CASIA	Number of training steps	10000	20000	30000	40000	50000
Before attack	Accuracy (%)	95.02	95.74	95.56	95.65	96.02
	HTER (%)	4.26	3.06	4.63	4.35	3.80
	Precision (%)	97.47	99.01	96.42	95.23	97.15
	Recall (%)	92.59	92.41	94.63	96.11	94.81
After attack	Recall (%)	16.85	22.78	20.00	28.15	27.41
	Recall drop (%)	75.74	69.63	74.63	67.96	67.40

CASIA dataset, on which a recall value of 16.85% is obtained when the model is trained using 10,000 training steps and increases to 27.41% at 50,000 steps.

### 4.2.5 Variation of fitness values

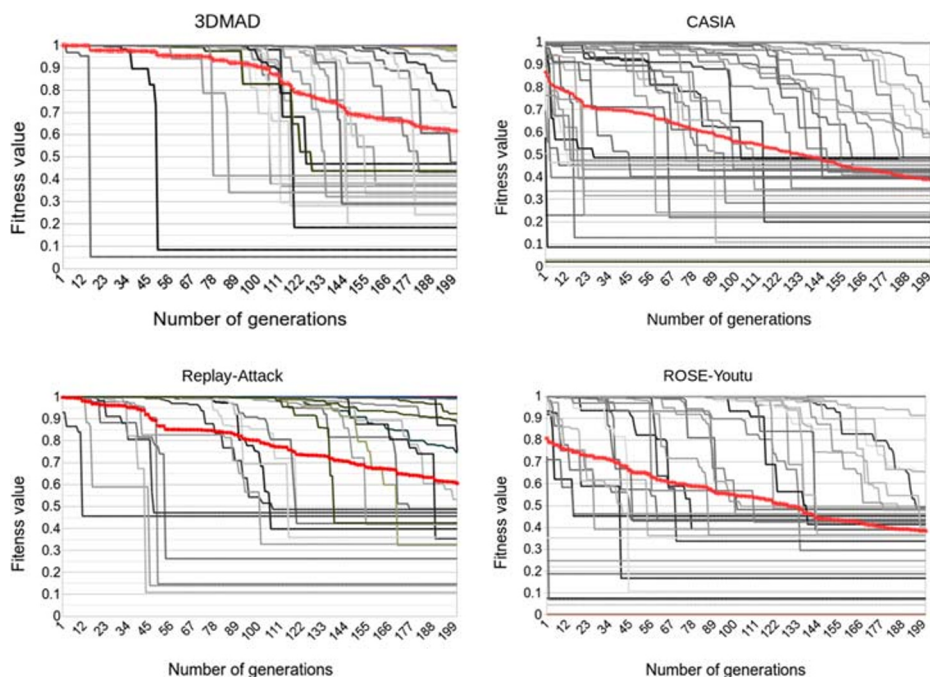
More experiments are carried out over 40 randomly selected fake images of each dataset to assess the differential evolution-based adversarial attack convergence. The adversarial attack is applied by manipulating 9 pixels for 200 attack iterations. The fitness values of the four datasets' evolution during the 200 attack iterations are illustrated in Figure 5, where red lines highlight the average values. Since the fitness values are set to be the original classes' confidence, the attack aims to decrease these values to reach 0.5 or lower. Once this value is achieved, the attack process ends, proving the constant fitness values under the 0.5 y-axis. Additionally, it can be noted from the figure that the average fitness value regularly decreases with the number of generations, which proves that the adversarial attack works as expected for most of the attacked images.

## 5 Conclusion

In this paper, we presented the results of an analytical study involving the application of differential evolution-based adversarial attacks to a VGG16-based face liveness detection model. The main goal of this study was to highlight practical criteria that can be used

**Table 8** Recall variation at different numbers of training steps for the anti-spoofing model on the Replay-Attack dataset. The numbers of manipulated pixels and attack iterations are held constant at nine and 200, respectively

Replay-Attack	Number of training steps	2500	10000	20000	30000	40000	50000
Before attack	Accuracy (%)	99.12	100.00	100.00	99.94	99.94	100.00
	HTER (%)	1.00	0.00	0.00	0.00	0.00	0.00
	Precision (%)	99.00	100.00	100.00	99.88	99.88	100.00
	Recall (%)	99.25	100.00	100.00	100.00	100.00	100.00
After attack	Recall (%)	50.75	56.88	64.75	55.00	41.75	66.38
	Recall drop (%)	48.50	43.12	35.25	45.00	58.25	33.62



**Fig. 5** Fitness values changes during 200 attack iterations among the four different face anti-spoofing face datasets

to improve defense strategies for deep learning-based anti-spoofing systems against such attacks. To do this, we conducted a series of experiments under different use-case scenarios to analyze the trade-off between adversarial attack algorithms and anti-spoofing systems. The use-case scenarios applied eight DA techniques and varied the number of training steps used to generate the anti-spoofing models, the number of manipulated pixels used in the attack, and the number of attack iterations. The results of experiments conducted on four face anti-spoofing databases revealed that training a model using more steps does not always reduce the recall drop. In addition, we found that increasing the number of pixels manipulated by the adversarial attack leads to a more significant recall drop and that DA could notably improve the VGG16-based anti-spoofing model's performance on the more complex datasets. However, we did not find that data augmentation can effectively protect the models from the adversarial attacks.

**Acknowledgments** The authors thank the Deanship of Scientific Research and RSSU at King Saud University for their technical support.

## References

1. Atoum Y, Liu Y, Jourabloo A, Liu X (2017) Face anti-spoofing using patch and depth-based CNNs. In: IEEE International Joint Conference on Biometrics (IJCB), pp 319–328
2. Bose AJ, Aarabi P (2018) Adversarial attacks on face detectors using neural net based constrained optimization. arXiv:[1805.12302](https://arxiv.org/abs/1805.12302)




3. Botelho de Souza G, Papa JP, Marana AN (2018) On the learning of deep local features for robust face spoofing detection. arXiv:[180607492](#)
4. Boulkenafet Z, Komulainen J, Li L, Feng X, Hadid A (2017) OULU-NPU: A mobile face presentation attack database with real-world variations. In: 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017), pp 612–618
5. Carlini N, Wagner D (2017) Towards evaluating the robustness of neural networks. arXiv:[160804644v2](#)
6. Chen X, Liu C, Li B, Lu K, Song D (2017) Targeted backdoor attacks on deep learning systems using data poisoning. arXiv:[v](#)
7. Chingovska I, Anjos A, Marcel S (2012) On the effectiveness of local binary patterns in face anti-spoofing. In: International Conference of Biometrics Special Interest Group (BIOSIG), pp 1–7
8. Chingovska I, Yang J, Le iZ, Yi D, Li S, Kahm O, Glaser C, Damer N, Kuijper A, Nouak A, Komulainen J, Pereira T, Gupta S, Khandelwal S, Bansal S, Rai A, Krishna T, Goyal D, Waris MA, Zhang H, Ahmad I, Kiranyaz S, Gabbouj M, Tronci R, Pili M, Sirena N, Roli F, Galbally J, Ficrrecz J, Pinto A, Pedrini H, Schwartz W, Rocha A, Anjos A, Marcel S (2013) The 2nd competition on counter measures to 2d face spoofing attacks. In: IAPR Int. Conference on Biometrics (ICB), pp 1–6
9. Dong Y, Liao F, Pang T, Su H, Zhu J, Hu X, Li J (2018) Boosting adversarial attacks with momentum. arXiv:[171006081](#)
10. Erdogmus N, Marcel S (2013) Spoofing in 2D face recognition with 3d masks and anti-spoofing with kinect. In: IEEE Sixth international conference on biometrics: Theory, Applications and Systems (BTAS), pp 1–6
11. Feng L, Po LM, Li Y, Xu X, Yuan F, Chun-HoCheung T, Cheung KW (2016) Integration of image quality and motion cues for face anti-spoofing: a neural network approach. *J Vis Commun Image Represent* 38:451–460
12. Fourati E, Elloumi W, Chetouani A (2020) Anti-spoofing in face recognition-based biometric authentication using image quality assessment. *Multimed Tools Appl* 79:865–889
13. Gan J, Li S, Zhai Y, Liu C (2017) 3D convolutional neural network based on face anti-spoofing. In: 2nd International Conference on Multimedia and Image Processing (ICMIP), pp 1–5
14. Goodfellow IJ, Shlens J, Szegedy C (2015) Explaining and harnessing adversarial examples. arXiv:[14126572](#)
15. Goswami G, Agarwal A, Ratha N, Singh R, Vatsa M (2019) Detecting and mitigating adversarial perturbations for robust face recognition. *Int J Comput Vis (IJCV)* 127:719–742
16. Hadid A (2014) Face biometrics under spoofing attacks: Vulnerabilities, countermeasures, open issues, and research directions. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp 113–118
17. Hu W, Te G, He J, Chen D, Guo Z (2018) Exploring hypergraph representation on face anti-spoofing beyond 2D attacks. arXiv:[181111594](#)
18. IJCB 2017 competition on generalized face presentation attack detection in mobile authentication scenarios. <https://sites.google.com/site/faceantispoofing/>, accessed 26 August 2019 (2017)
19. ISO/IEC 30107-1:2016 information technology biometric presentation attack detection part 1: Framework. <https://www.iso.org/standard/53227.html>, accessed 26 August 2019 (2016)
20. Jiawei SVDV, Sakurai K (2019) Attacking convolutional neural network using differential evolution. *IPSP Trans Comput Vis Appl* 11:1–12
21. Jourabloo A, Liu Y, Liu X (2018) Face de-spoofing: Anti-spoofing via noise modeling. arXiv:[180709968](#)
22. Komulainen J (2015) Software-based countermeasures to 2D facial spoofing attacks. PhD thesis, University of Oulu
23. Kose N, Dugelay JL (2013) On the vulnerability of face recognition systems to spoofing mask attacks. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp 2357–2361
24. Krizhevsky A, Sutskever I, Hinton GE (2017) Imagenet classification with deep convolutional neural networks. *Adv Neural Inf Process Syst (NIPS)* 60:1097–1105
25. Kurakin A, Goodfellow I, Bengio S (2017a) Adversarial examples in the physical world. arXiv:[160702533v4](#)
26. Kurakin A, Goodfellow I, Bengio S (2017b) Adversarial machine learning at scale. arXiv:[161101236v2](#)
27. Li L, Feng X, Boulkenafet Z, Xia Z, Li M, Hadid A (2016) An original face anti-spoofing approach using partial convolutional neural network. In: Sixth International Conference on Image Processing Theory, Tools and Applications (IPTA), pp 1–6
28. Li L, Correia PL, Hadid A (2018) Face recognition under spoofing attacks: countermeasures and research directions. *IET Biom* 7:3–14
29. Li H, Li W, Cao H, Wang S, Huang F, Kot AC (2018) Unsupervised domain adaptation for face anti-spoofing. *IEEE Transactions on Information Forensics and Security* 13

30. Liu Y, Jourabloo A, Liu X (2018) Learning deep models for face anti-spoofing: Binary or auxiliary supervision. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp 389–398
31. Liu Y, Stehouwer J, Jourabloo A, Liu X (2019) Deep tree learning for zero-shot face anti-spoofing. arXiv:190402860
32. Liu Y, Tai Y, Li J, Ding S, Wang C, Huang F, Li D, Qi W, Ji R (2019) Aurora guard: Real-time face anti-spoofing via light reflection. arXiv:190210311
33. Liu A, Wan J, Escalera S, Escalante HJ, Tan Z, Yuan Q, Wang K, Lin C, Guo G, Guyon I, Li SZ (2019) Multi-modal face anti-spoofing attack detection challenge at CVPR2019. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops
34. Lucena O, Junior A, Moia V, Souza R, Valle E, Lotufo R (2017) Transfer learning using convolutional neural networks for face anti-spoofing. In: International Conference Image Analysis and Recognition (ICIAR), pp 27–34
35. Moosavi-Dezfooli SM, Fawzi A, Fawzi O, Frossard P (2017) Universal adversarial perturbations. arXiv:161008401
36. Moosavi-Dezfooli SM, Fawzi A, Frossard P (2016) Deepfool: a simple and accurate method to fool deep neural networks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 2574–2582
37. Nagpal C, Dubey SR (2019) A performance evaluation of convolutional neural networks for face anti spoofing. arXiv:180504176
38. Nguyen MD, Bui QM (2009) Your face is NOT your password. In: Black hat DC
39. Papernot N, McDaniel P, Goodfellow IJ, Jha SK, Celik ZB, Swami A (2017) Practical black-box attacks against machine learning. In: ACM on Asia Conference on Computer and Communications Security, pp 506–519
40. Patel K, Han H, Jain AK (2016) Secure face unlock: Spoof detection on smartphones. IEEE Trans Inf Forens Secur 11:2268–2283
41. Patel K, Han H, Jain AK (2016) Cross-database face antispoofing with robust feature representation. In: Chinese Conference on Biometric Recognition (CCBR), pp 611–619
42. Peng F, Qin L, Long M (2018) Face presentation attack detection using guided scale texture. Multimed Tools Appl 77:1–27
43. Redmon J, Farhadi A (2016) YOLO9000: Better, faster, stronger. arXiv:161208242
44. Saon G, Kuo HKJ, Rennie S, Picheny M (2015) The IBM 2015 English conversational telephone speech recognition system. arXiv:150505899
45. Sharif M, Bhagavatula S, Bauer L, Reiter MK (2016) Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. In: ACM SIGSAC Conference on Computer and Communications Security, pp 1528–1540
46. Sharif M, Bhagavatula S, Bauer L, Reiter MK (2019) A general framework for adversarial examples with objectives. arXiv:180100349
47. Simonyan K, Zisserman A (2015) Very deep convolutional networks for large-scale image recognition. arXiv:14091556
48. Song Q, Wu Y, Yang L (2018) Attacks on state-of-the-art face recognition using attentional adversarial attack generative network. arXiv:181112026
49. Souza L, Oliveira L, Pamplona M, Papa JP (2018) How far did we get in face spoofing detection? Eng Appl Artif Intell 72:368–381
50. Storn R, Price K (1997) Differential evolution a simple and efficient heuristic for global optimization over continuous spaces. J Glob Optim 11:341–359
51. Su J, Vargas DV, Kouichi S (2018) One pixel attack for fooling deep neural networks. arXiv:171008864
52. Szegedy C, Zaremba W, Sutskever I, Bruna J, Erhan D, Goodfellow I, Rob F (2014) Intriguing properties of neural networks. ICLR, arXiv:13126199
53. Trusted biometrics under spoofing attacks (TABULA RASA). <http://www.tabularasa-euproject.org/>, accessed 26 August 2019 (2010)
54. Tu X, Zhang H, Xie M, Luo Y, Zhang Y, Ma Z (2019) Deep transfer across domains for face anti-spoofing. arXiv:190105633
55. Tu X, Zhang H, Xie M, Luo Y, Zhang Y, Ma Z (2019) Enhance the motion cues for face anti-spoofing using CNN-LSTM architecture. arXiv:190105635
56. Unique identification authority of India (UIDAI). <https://uidai.gov.in/>, accessed 26 August 2019 (2016)
57. Ur Rehman YA, Po LM, Liu M, Zou Z, Ou W, Zhao Y (2019) Face liveness detection using convolutional-features fusion of real and deep network generated face images. J Vis Commun Image Represent 59:574–582

58. Wang Z, Zhao C, Qin Y, Zhou Q, Qi G, Wan J, Lei Z (2019) Exploiting temporal and depth information for multi-frame face anti-spoofing. arXiv:181105118v3
59. Wen D, Han H, Jain AK (2015) Face spoof detection with image distortion analysis. *IEEE Trans Inf Forens Secur* 10:746–761
60. Yang J, Lei Z, Li SZ (2014) Learn convolutional neural network for face anti-spoofing. *Computer Science*. arXiv:1408.5601, pp 373–384
61. Zhang Z, Yan J, Liu S, Lei Z, Yi D, Li SZ (2012) A face antispoofing database with diverse attacks. In: 5th IAPR International Conference on Biometrics (ICB), pp 26–31
62. Zhang K, Zhang Z, Li Z, Qiao Y (2016) Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Process Lett* 23:1499–1503
63. Zhou Z, Tang D, Wang X, Han W, Liu X, Zhang K (2018) Invisible mask: Practical attacks on face recognition with infrared. arXiv:180304683

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

## Affiliations

Naima Bousnina<sup>1</sup>  · Lilei Zheng<sup>2</sup> · Mounia Mikram<sup>3</sup> · Sanaa Ghouzali<sup>4</sup> · Khalid Minaoui<sup>1</sup>

Lilei Zheng  
lilei.zheng@shopee.com

Mounia Mikram  
mikrammounia@gmail.com

Sanaa Ghouzali  
sghouzali@ksu.edu.sa

Khalid Minaoui  
kminaoui@gmail.com

- <sup>1</sup> LRIT - CNRST URAC n°. 29, Faculty of Sciences Rabat, IT Center, Mohammed V University Morocco, Rabat, Morocco
- <sup>2</sup> Image Processing Team, Data Science, Shopee Singapore, Singapore, Singapore
- <sup>3</sup> Meridian Team, LYRICA Laboratory, School of Information Sciences, Rabat, Morocco
- <sup>4</sup> Information Technology Department, College of Computer and Information Sciences, King Saud University Riyadh, Riyadh, Kingdom of Saudi Arabia